Building Physic
KU Leuven & Energy Ville

EnergyVille

A Library for Linear Modelling of Energy Systems

9 novembre 2017
Vincent reinbold
1 - Introduction
- GeoWatt : Context and Research Directions
- Scientific Issues
- State of the art and Proposal

2 - Library for Linear Modelling of Energy System
- Introduction
- LLMSE : Library for Linear Modelling of Energy Systems
- A Practical Implementation

3 - Conclusion
1 - Introduction

- GeoWatt: Context and Research Directions
- Scientific Issues
- State of the art and Proposal

2 - Library for Linear Modelling of Energy System

- Introduction
- LLMSE: Library for Linear Modelling of Energy Systems
- A Practical Implementation

3 - Conclusion
**Context and Benefits**

- **Grid & Transp.**
- **Storage**
- **Microgrid**
- **Building**
- **Renewable Generation**
- **Market**

**Research Directions**

- Research on microgrids and smart buildings
  - Optimization of energy demand
  - Solutions for generation
  - Storage management
  - Economic feasibility
  - Environmental assessments

---

*GeoWatt: Context and Research Directions*
Outline

1 - Introduction
- GeoWatt: Context and Research Directions
- Scientific Issues
- State of the art and Proposal

2 - Library for Linear Modelling of Energy System
- Introduction
- LLMSE: Library for Linear Modelling of Energy Systems
- A Practical Implementation

3 - Conclusion
Topography, Energy Management and Sizing Optimization
What are the Problems?

- **Real Time Control**
- **Energy Management**: Unit Commitment
- **Continuous Sizing**: Sections, Capacities, Nominal Values, etc.
- **Topology Optimization**: Network, Storage Position, etc.
**What are the Problems?**

- **Real Time Control**
- **Energy Management**: Unit Commitment
- **Continuous Sizing**: Sections, Capacities, Nominal Values, etc.
- **Topology Optimization**: Network, Storage Position, etc.

Subjected to Uncertainties
### What are the Problems?

- **Real Time Control**
- **Energy Management**: Unit Commitment
- **Continuous Sizing**: Sections, Capacities, Nominal Values, etc.
- **Topology Optimization**: Network, Storage Position, etc.

### Minimize Cost, subjected to:

- **Forecasts**: generation, loads, occupancy, weather
- **Informations**: GIS, BIM, measurements
- Energy balance,
- **Load/Storage Management**,
- **Physical limits**: the models
### Time and Space Scales

#### Time Scale
- Generation Following
- Daily Management
- Seasonal Study
- Life Cycle Analysis

#### Space Scale
- **Building**: Loads Control, DHW and envelop Sizing
- **District Scale**: Aggregate Loads, Storage, Local Generation
- **City/Region Scale**: Network, Generation
Time and Space Scales

**Time Scale**
- Generation Following
- Daily Management
- Seasonal Study
- Life Cycle Analysis

**Space Scale**
- **Building**: Loads Control, DHW and envelop Sizing
- **District Scale**: Aggregate Loads, Storage, Local Generation
- **City/Region Scale**: Network, Generation

![Time Scale Diagram](image-url)

60 x 60 x 24 x 30 x 12 x 20 ≈ 6,2.10^8
1 - Introduction
- GeoWatt: Context and Research Directions
- Scientific Issues
- State of the art and Proposal

2 - Library for Linear Modelling of Energy System
- Introduction
- LLMSE: Library for Linear Modelling of Energy Systems
- A Practical Implementation

3 - Conclusion
State of the Art

What are the Approaches?

- Network scheduling and sizing $\rightarrow$ Optimization
- Approaches
- Available methods
- Models
State of the Art

What are the Approaches?

- Network scheduling and sizing → Optimization

Approaches
- Fully-Centralized
- Aggregations, Multi-level
- Fully-Distributed (decomposition)

Available methods

Models
State of the Art

What are the Approaches?

Network scheduling and sizing → Optimization

Approaches

Available methods

- Deterministic
- Heuristic
- Rule-based
- Hybrid

Models
State of the Art

What are the Approaches?

- Network scheduling and sizing → Optimization
- Approaches
- Available methods
- Models
  - LP, MILP
  - QP, QCQP, SDP
  - NLP, MINLP
  - Stochastic,

Linear & Mixed Integer Programming

- Modelling physic
- Convergence propriety, problem size

Non-Linear Programming

- Convergence speed/quality
- Time Consuming Simulation, Jacobians, etc.
- Accurate Modeling,

Stochastic

- Modeling, Convergence, Speed
- Uncertainties, Robust
State of the Art

What are the Approaches?

- Network scheduling and sizing → Optimization
- Approaches
- Available methods
- Models

Linear & Mixed Integer Programming
- Modelling physics
- Convergence propriety, problem size

Non-Linear Programming
- Convergence speed/quality
- Time Consuming Simulation, Jacobians, etc.
- Accurate Modeling,

Stochastic
- Modeling, Convergence, Speed
- Uncertainties, Robust
Proposal - One Tool to Gather and Optimize DES

Main Objective

Develop an Oriented Object python package for the Mixed-Integer Linear Modeling of District Energy Systems.

- Linear & Mixed Integer Linear (& Quadratic) Programming
  - Convergence, Speed, Matrix size
  - Energy Management, Topology and Sizing Problems,
  - Compatible with Centralized, Aggregative or Decentralize Methods,
    Stochastic/Robust optimization (hybrid).

- Object Oriented (Python and solver API)
  - To make the modeling and post-processing easier (vs. GAMS, AMPL, etc.)
  - To gather and share models

- Multi-Physic Modeling
  - Thermal Building Structure, Network, Storage
  - Electrical Grid Connection, co-generation, power-to-gaz
  - Fluid-Mechanics Pipes, DHW, Substations, etc.
Proposal - One Tool to Gather and Optimize DES

Main Objective

Develop an Oriented Object python package for the Mixed-Integer Linear Modeling of District Energy Systems.

- **Linear & Mixed Integer Linear (& Quadratic) Programming**
  - Convergence, Speed, Matrix size
  - Energy Management, Topology and Sizing Problems,
  - Compatible with Centralized, Aggregative or Decentralize Methods, Stochastic/Robust optimization (hybrid).

- **Object Oriented (Python and solver API)**
  - To make the modeling and post-processing easier (vs. GAMS, AMPL, etc.)
  - To gather and share models

- **Multi-Physic Modeling**
  - **Thermal** Building Structure, Network, Storage
  - **Electrical** Grid Connection, co-generation, *power-to-gaz*
  - **Fluid-Mechanics** Pipes, DHW, Substations, etc.
Proposal - One Tool to Gather and Optimize DES

Main Objective

Develop an Oriented Object python package for the Mixed-Integer Linear Modeling of District Energy Systems.

- **Linear & Mixed Integer Linear (& Quadratic) Programming**
  - Convergence, Speed, Matrix size
  - Energy Management, Topology and Sizing Problems,
  - Compatible with Centralized, Aggregative or Decentralize Methods, Stochastic/Robust optimization (hybrid).

- **Object Oriented (Python and solver API)**
  - To make the modeling and post-processing easier (vs. GAMS, AMPL, etc.)
  - To gather and share models

- **Multi-Physic Modeling**
  - Thermal Building Structure, Network, Storage
  - Electrical Grid Connection, co-generation, power-to-gaz
  - Fluid-Mechanics Pipes, DHW, Substations, etc.
Proposal - One Tool to Gather and Optimize DES

Main Objective

Develop an Oriented Object python package for the Mixed-Integer Linear Modeling of District Energy Systems.

- Linear & Mixed Integer Linear (& Quadratic) Programming
  - Convergence, Speed, Matrix size
  - Energy Management, Topology and Sizing Problems,
  - Compatible with Centralized, Aggregative or Decentralize Methods, Stochastic/Robust optimization (hybrid).

- Object Oriented (Python and solver API)
  - To make the modeling and post-processing easier (vs. GAMS, AMPL, etc.)
  - To gather and share models

- Multi-Physic Modeling
  - Thermal Building Structure, Network, Storage
  - Electrical Grid Connection, co-generation, power-to-gaz
  - Fluid-Mechanics Pipes, DHW, Substations, etc.
Outline

1 - Introduction
  - GeoWatt: Context and Research Directions
  - Scientific Issues
  - State of the art and Proposal

2 - Library for Linear Modelling of Energy System
  - Introduction
  - LLMSE: Library for Linear Modelling of Energy Systems
  - A Practical Implementation

3 - Conclusion
Definitions

**Integer Program**

An optimization model is an Integer Program if any of its decision variables is discrete

- If all variables are discrete, the model is a pure integer program
- Otherwise, the model is a mixed-integer program

**Standard Mixed-Integer Linear Programming (MILP) Formulation**

\[
\min_{x,y} \quad c^\top x + d^\top y \\
\text{s.t.} \quad Ax + Ey \begin{cases} 
\geq \\
= \\
\leq \end{cases} b \\
x_{\min} \leq x \leq x_{\max} \\
y \in 0; 1^{ny}
\]
Definitions

Linear vs. Non-Linear Programming

- An IP model is an Integer linear program (ILP) if its (single) objective function and all its constraints are linear.
- Otherwise, it is an integer nonlinear program (INLP).

Standard Mixed-Integer Linear Programming (MILP) Formulation

\[ \min_{x,y} \quad c^T x + d^T y \]

subject to

\[ Ax + Ey \begin{cases} \geq & \quad \geq b \\ = & \quad = \end{cases} \]

\[ x_{\text{min}} \leq x \leq x_{\text{max}} \]

\[ y \in 0; 1^{ny} \]
**Definitions**

### Linear vs. Non-Linear Programming

- An IP model is an Integer linear program (ILP) if its (single) objective function and all its constraints are linear.
- Otherwise, it is an integer nonlinear program (INLP).

### Standard Mixed-Integer Linear Programming (MILP) Formulation

\[
\begin{align*}
\min_{x,y} & \quad c^T x + d^T y \\
\text{s.t.} & \quad A x + E y \begin{cases} \geq \end{cases} b \\
& \quad \begin{cases} = \end{cases} \begin{cases} \leq \end{cases} \\
& \quad x_{\min} \leq x \leq x_{\max} \\
& \quad y \in 0; 1^{n_y}
\end{align*}
\]
The concept

Basic Ideas

- The **Unit** integrates his own **Quantities**, **Constraints** and **Objectives**
- **Connexions** between Units create **Global Constraints**
- The Problem is created by **aggregating Units** and **Connecting** them
Outline

1 - Introduction
- GeoWatt: Context and Research Directions
- Scientific Issues
- State of the art and Proposal

2 - Library for Linear Modelling of Energy System
- Introduction
- LLMSE: Library for Linear Modelling of Energy Systems
- A Practical Implementation

3 - Conclusion
Quick View to the Source

The Source:

Library For Linear Modeling of Energetic Systems, rtd: https://reinboldv.github.io/llmse/

- python-library
- python
- optimization
- modelling
- Manage topics

134 commits
5 branches
3 releases
2 contributors

Branch: gh-pages

- Examples: tutorial step1 and 2 finished
- docs: docs modif
- llmse: hot fix python 3.6
- .gitattributes: change versioneer options
- .gitignore: update in ocre, thermal and template for tutorial (not finished)
- .nojekyll: Create .nojekyll
- MANIFEST.in: update examples and delete old architecture
- README.rst: modification documentation

Vincent reinbold
Quick View to the Documentation

Welcome to the LLMES’s documentation!

This is the documentation of the LLMES’s python package (Library For Linear Modeling of Energetic Systems). This project is about the mixed linear modeling of energetic systems in python using gurobipy and SciPy packages. Gurobipy is a wrapper allowing to model and solve mixed integer linear programming within python language. More information here: Gurobi Python API Overview.

Getting Started

- Installation steps
  - Installing Python distribution (Anaconda)
  - Install LLMSE from source
  - Install Gurobi Solver
  - Install dependencies
- How to build the documentation?
Outline

1 - Introduction
   - GeoWatt: Context and Research Directions
   - Scientific Issues
   - State of the art and Proposal

2 - Library for Linear Modelling of Energy System
   - Introduction
   - LLMSE: Library for Linear Modelling of Energy Systems
   - A Practical Implementation

3 - Conclusion
A Practical Implementation

1. **Instantiation of the Model**

   ```python
   # time horizon and time step definition
   th = Time(end=23, freq='H')
   # Model instantiation
   mgm = MGModel(name='Smart Building Example')
   # Thermal envelope instanciation
   bui = SingleZoneBuilding(th, name='BUI0', ...)
   # Battery instantiation
   sb0 = SimpleBattery(th, name='SB0', emax=10, emin=0, pcmax=10, pdmax=10)
   # Wind Turbine instantiation
   wt0 = WindTurbine(th, name='WT0', ...)
   # Main Grid Connection
   mg0 = MainGridt(th, name='MG0', pmax=20, pmin=20, cout=cout, cin=cin)
   ```

2. Aggregation of variables, constraints and objectives

3. Multi-physic connections between units

4. Optimization
A Practical Implementation

1. **Instantiation of the Model**

```python
1 # time horizon and time step definition
2 th = Time(end=23, freq='H')
3 # Model instantiation
4 mgm = MGModel(name='Smart—Building Example')
5 # Thermal envelope instanciation
6 bui = SingleZoneBuilding(th, name='BUI0', ...)
7 # Battery instantiation
8 sb0 = SimpleBattery(th, name='SB0', emax=10, emin=0,
9    pcmax=10, pdmax=10)
10 # Wind Turbine instantiation
11 wt0 = WindTurbine(th, name='WT0', ...)
12 # Main Grid Connection
13 mg0 = MainGrid(th, name='MG0', pmax=20, pmin=20,
14    cout=cout, cin=cin)
```

2. **Aggregation of variables, constraints and objectives**

3. **Multi-physic connections between units**

4. **Optimization**
1. **Instantiation of the Model**

```python
1 # time horizon and time step definition
2 th = Time(end=23, freq='H')
3 # Model instantiation
4 mgm = MGModel(name='Smart Building Example')
5 # Thermal envelope instantiation
6 bui = SingleZoneBuilding(th, name='BUI0', ... )
7 # Battery instantiation
8 sb0 = SimpleBattery(th, name='SB0', emax=10, emin=0,
9    pcmax=10, pdmax=10)
10 # Wind Turbine instantiation
11 wt0 = WindTurbine(th, name='WT0', ...)
12 # Main Grid Connection
13 mg0 = MainGridt(th, name='MG0', pmax=20, pmin=20,
14    cout=cout, cin=cin)
```

2. **Aggregation of variables, constraints and objectives**

3. **Multi-physic connections between units**

4. **Optimization**
1. Instantiation of the Model

1. # time horizon and time step definition
2. th = Time(end=23, freq='H')
3. # Model instantiation
4. mgm = MGModel(name='Smart—Building Example')
5. # Thermal envelope instanciation
6. bui = SingleZoneBuilding(th, name='BUI0', ...)
7. # Battery instantiation
8. sb0 = SimpleBattery(th, name='SB0', emax=10, emin=0,
9. pcmax=10, pdmax=10)
10. # Wind Turbine instantiation
11. wt0 = WindTurbine(th, name='WT0', ...)
12. # Main Grid Connection
13. mg0 = MainGridt(th, name='MG0', pmax=20, pmin=20,
14. cout=cout, cin=cin)

2. Aggregation of variables, constraints and objectives

3. Multi-physic connections between units

4. Optimization
A Practical Implementation

1. Instantiation of the Model
2. Aggregation of variables, constraints and objectives

```c
mgm.addunit(th, bui, sb0, wt0, mg0)
```

3. Multi-physic connections between units
4. Optimization
A Practical Implementation

1. Instantiation of the Model
2. Aggregation of variables, constraints and objectives
3. Multi-physic connections between units

   16. `mgm.addEffortConnection(...)` # introduce equality
   17. `mgm.addFluxConnection(...)` # introduce conservation equality

4. Optimization
A Practical Implementation

1. Instantiation of the Model
2. Aggregation of variables, constraints and objectives
3. Multi-physic connections between units
4. Optimization

18  mgm.update()
19  mgm.optimize()
A Practical Implementation

1. Instantiation of the Model
2. Aggregation of variables, constraints and objectives
3. Multi-physic connections between units
4. Optimization

Smart-Grid Example (24h)

- 19 lines of code (∼ 1000 in total) ≡ 4 000 lines in LP language
- 1 400 variables, 2 400 constraints
- Optimization time ≤ 1 s
  (i7-6600U, 2.60GHz, 8GB / Python 2.7 / Gurobi 6.5)
Conclusion - A General Tool for Modeling Optimization Problems

1. Oriented Object Tool for Optimization
   - Make the Optimization Formulation Easier
   - Develop, Share and Gather Models For DES
   - **Documentation** and **Post-Processing** Easier using Python

2. Application for District Energy System
1. Oriented Object Tool for Optimization
2. Application for District Energy System
   - (In development) Modeling of Pipes, Network, Thermal Storage
   - (In development) Social, Environmental and Economical models
   - Topology and Sizing Optimization
   - Comparison with NL models
Conclusion - A General Tool for Modeling Optimization Problems

1. Oriented Object Tool for Optimization
2. Application for District Energy System

Perspectives

- Develop/Feed Models and Examples for the Library,
- Building a community
- Real Implementation & Measurements